

(12) UK Patent Application (19) GB (11) 2 299 736 (13) A

(43) Date of A Publication 09.10.1996

(21) Application No 9606467.0

(22) Date of Filing 27.03.1996

(30) Priority Data

(31) 08418526

(32) 07.04.1995

(33) US

(71) Applicant(s)

Sony Electronics Inc.

(Incorporated in USA - Delaware)

**One Sony Drive, Park Ridge, New Jersey 07656,
United States of America**

(72) Inventor(s)

Oliver F Morgan

(74) Agent and/or Address for Service

D Young & Co

**21 New Fetter Lane, LONDON, EC4A 1DA,
United Kingdom**

(51) INT CL⁶

G06F 3/023

(52) UK CL (Edition O)

H4T TBLA T130 T143

(56) Documents Cited

GB 2254989 A EP 0570137 A1 EP 0468911 A2

US 5422993 A

(58) Field of Search

UK CL (Edition O) H4T TBLA TBLC TBLM TBLX

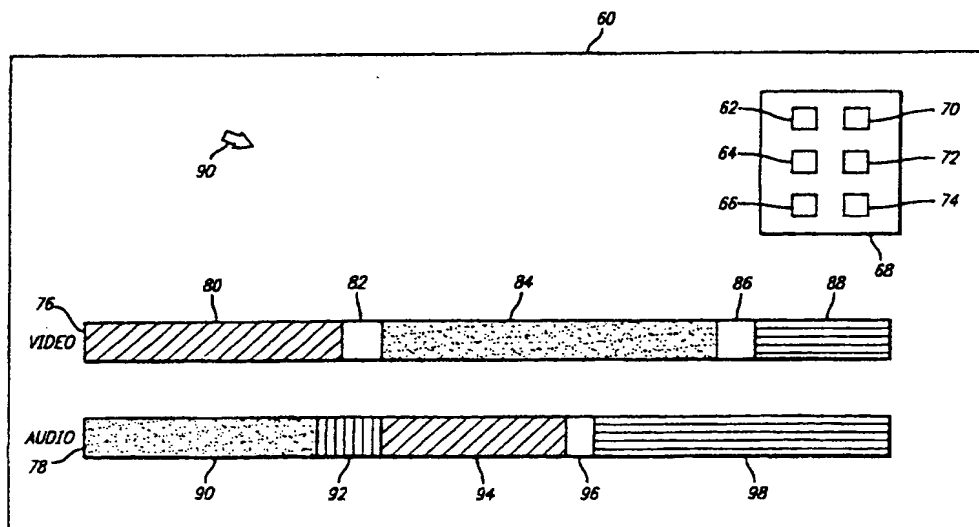
INT CL⁶ G06F 3/023 3/033 3/037

ONLINE: WPI;COMPUTER

(54) GRAPHICAL USER INTERFACES

(57) A graphical user interface for use in computer systems includes a function palette 68 that displays a variety of function icons 62 to 74. Each function icon corresponds to a function that a user may wish to perform. The user selects a function icon from the palette corresponding to the desired function and places a copy of the icon at that point in the display where the function is to be performed. For example, in the context of a multimedia video editor, the user may place an icon on a timeline 76, 78 that corresponds to source material. The desired function will then be performed on the source material at that point where the user placed the icon, either immediately or upon subsequent command from the user.

FIG. 2



GB 2 299 736 A

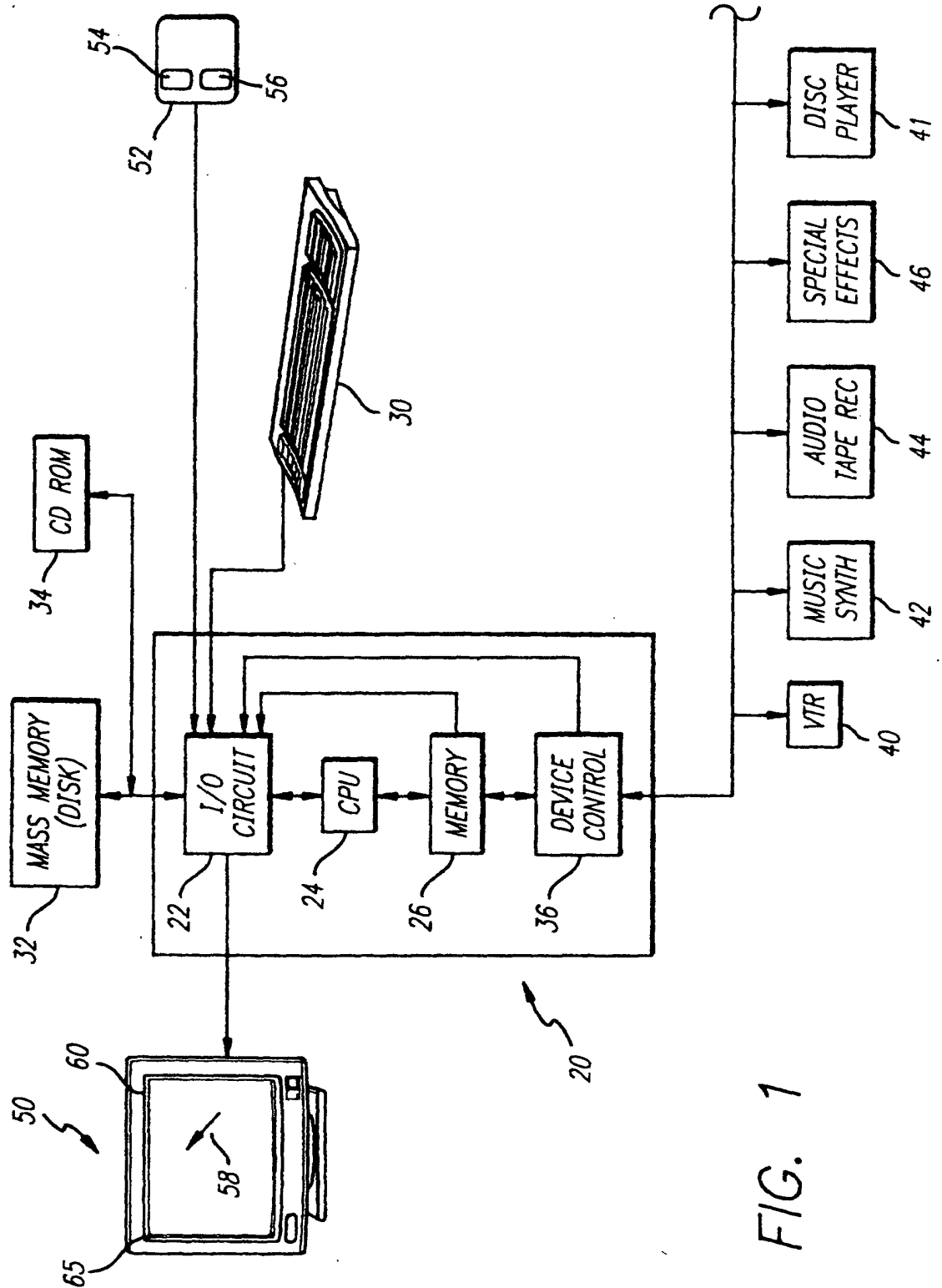
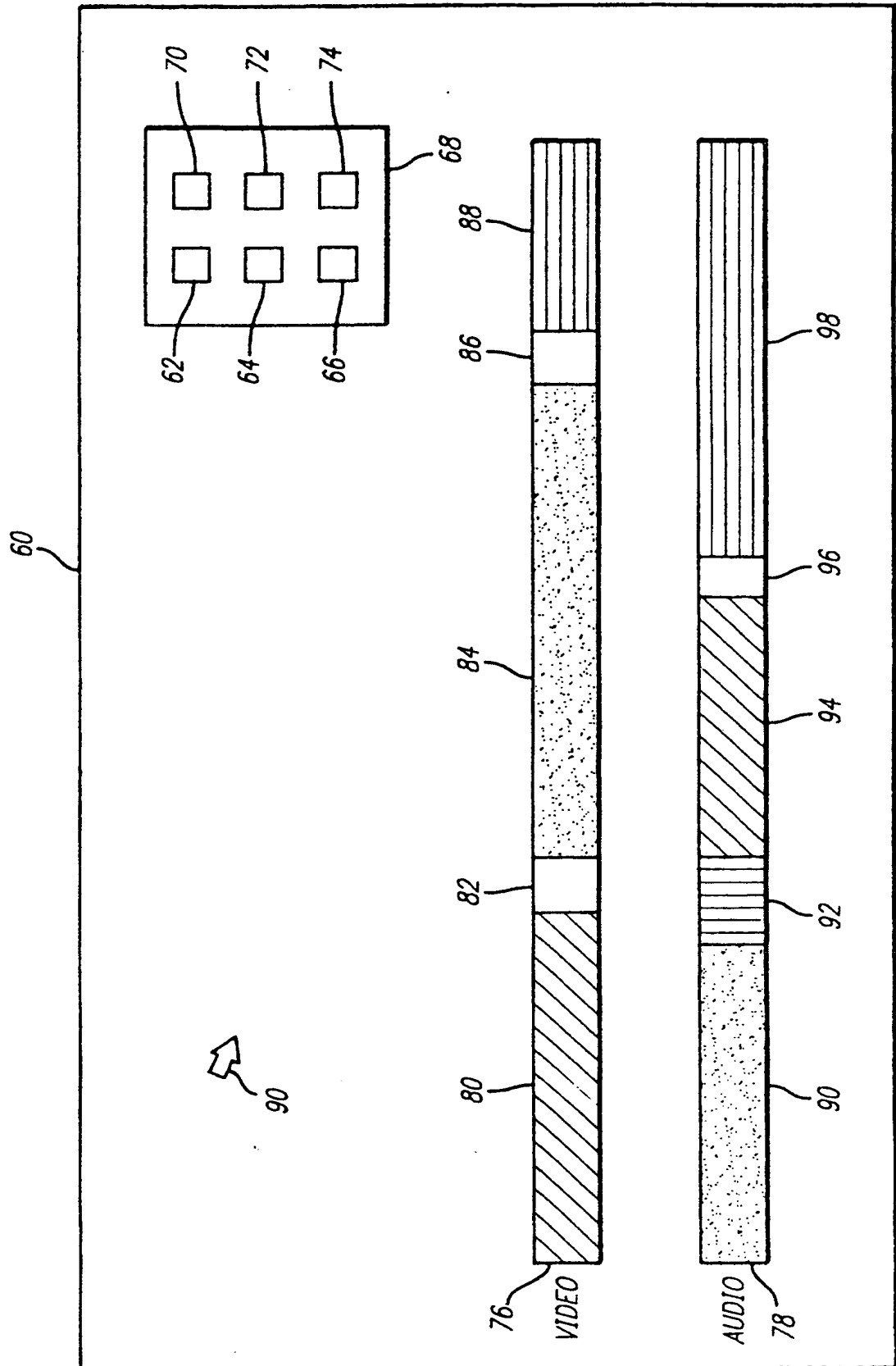


FIG. 1

FIG. 2



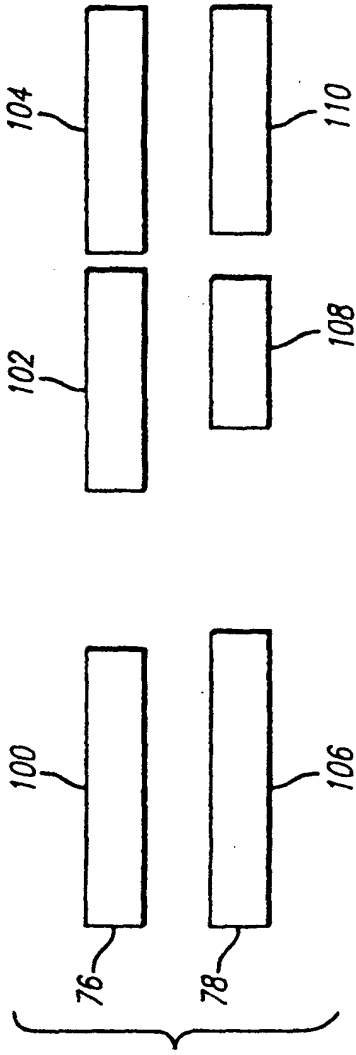


FIG. 3a

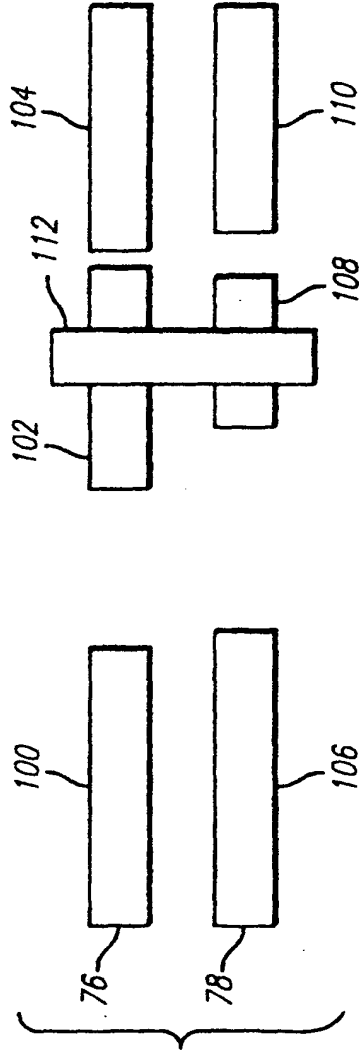


FIG. 3b

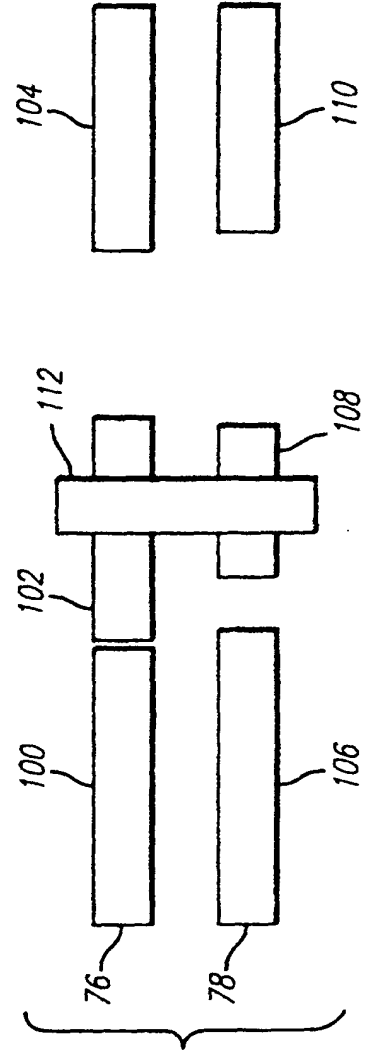


FIG. 3c

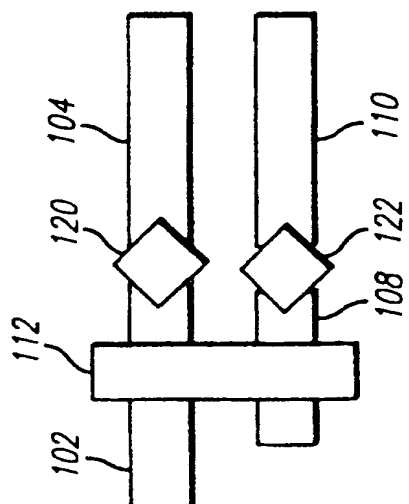


FIG. 4a

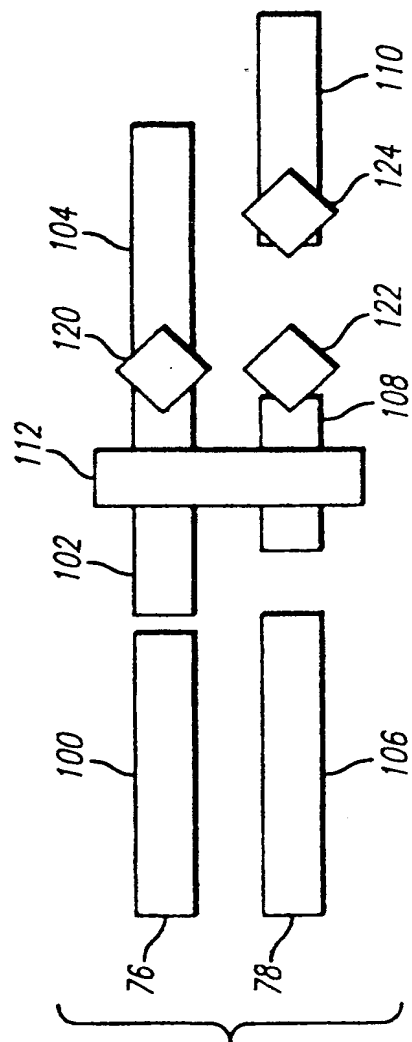


FIG. 4b

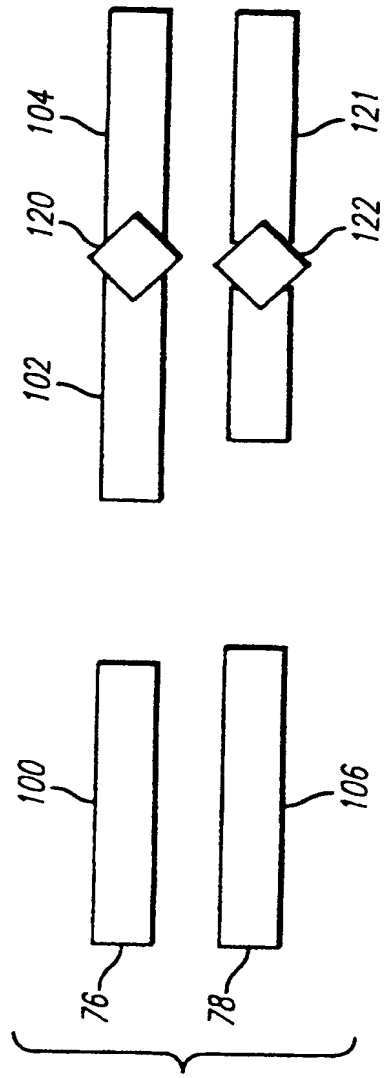


FIG. 5a

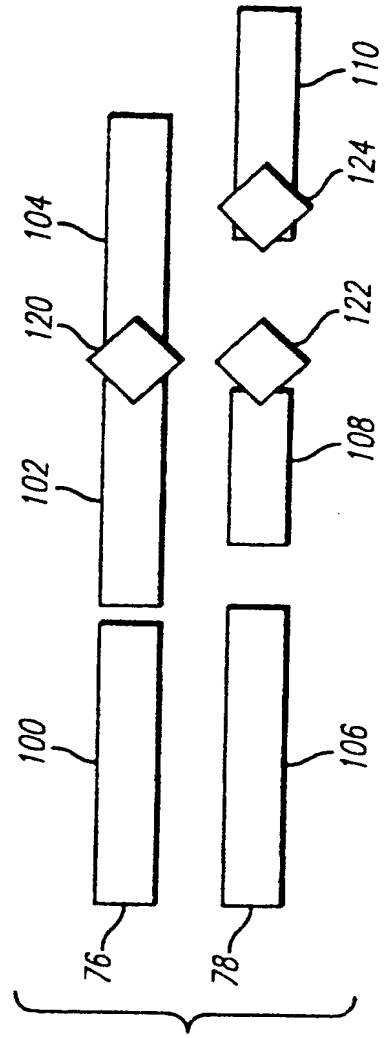


FIG. 5b

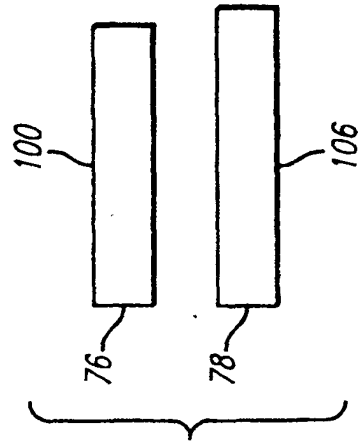
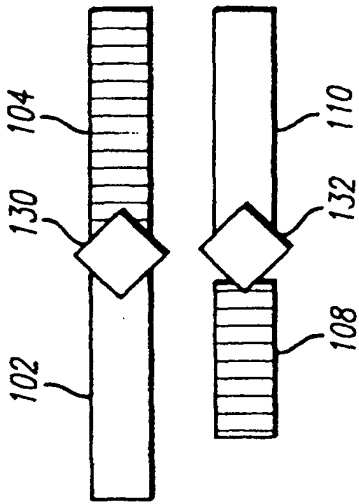


FIG. 6a

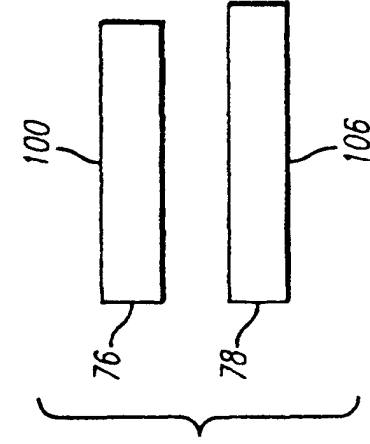
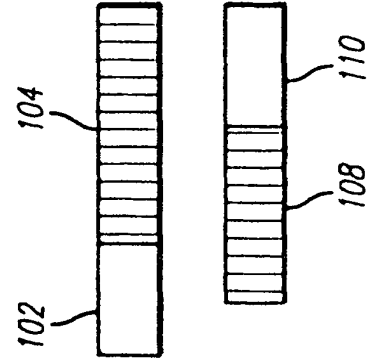


FIG. 6b

FIG. 7a

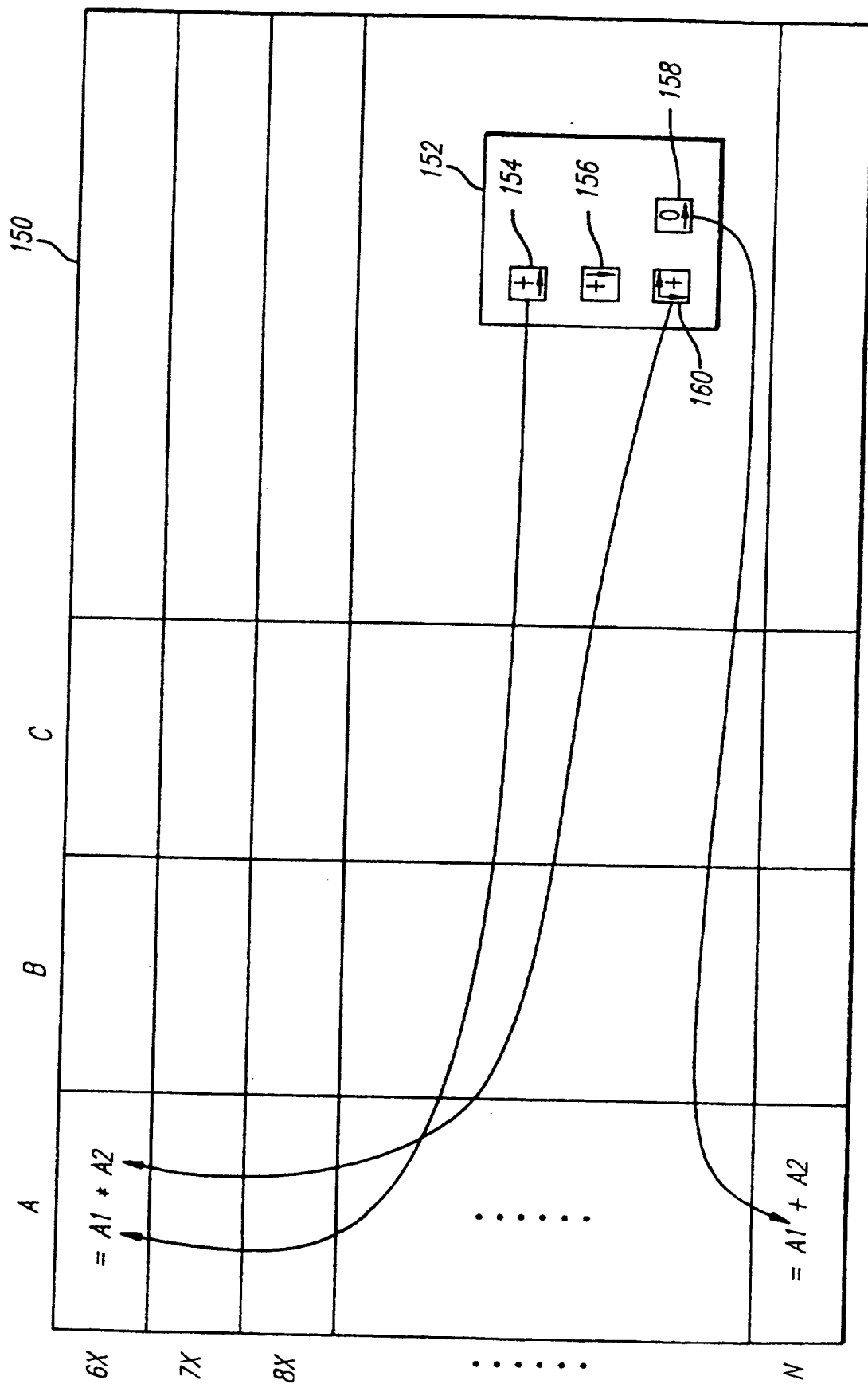


FIG. 7b

	A	B	C	
6X	$= A1 * A2$	$= B1 * B2$	$= C1 * C2$	
7X	$= A1 * A3$	$= B1 * B3$	$= C1 * C3$	
8X	$= A1 * A4$	$= B1 * B4$	$= C1 * C4$	
...	...			<div data-bbox="868 315 1250 661"> <p>Diagram 150 illustrates a set of three adders, labeled 154, 156, and 158, each represented by a square box containing a plus sign (+). These adders are arranged horizontally. Above the first adder (154) is a bracket labeled 152, which also encompasses the second adder (156). To the right of the third adder (158) is a square box containing the number 0, with an arrow pointing from the adder to this box. The entire diagram is enclosed in a larger rectangular frame.</p> </div>
N	$= A1 + A2$	$= BN + A2$		

GRAPHICAL USER INTERFACES

This invention relates to the field of displaying, manipulating, and/or selecting multimedia or computer stored information. More
5 particularly, the invention relates to the provision of graphical user interfaces on computer systems. Graphical user interfaces provided by embodiments of the invention can be used for a variety of applications, including but not limited to a multi-media video and audio editing system.

Over the past decade, a variety of graphical user interfaces have
10 been developed to ease human interaction with computer systems. Many display systems utilize metaphors in the design of the interface as a way of maximizing human familiarity, and conveying information between the user and the computer. It is common for computer systems incorporating so-called "object oriented" display systems to utilize multiple "windows"
15 on a display in which combinations of text and graphics are disposed. The display systems utilize familiar metaphors, such as desktops, control panels, notebooks and the like, such that the interface

takes advantage of existing human mental structures to permit a user to draw upon the metaphor analogy to understand the requirements of the particular computer system.

5 It is well known that designing around a familiar metaphor helps reduce human learning time. (See for example, Patrick Chan, "Learning Considerations in User Interface Design: The Room Model", Report CS-84-16, University of Waterloo Computer Science Department, Ontario, Canada, July, 1984, and the references cited therein.) The ability to operate on images or objects which relate
10 the user to actual objects on, for example, a desktop, results in a stronger man-machine interface. (See for example, D. Robson, "Object Oriented Software Systems", BYTE, August 1991, page 74, Vol. 6, No. 8; L. Tesler, "The Smalltalk Environment", BYTE, August 1981, page 90, Vol. 6, No. 8; and Smith, et al., "Star User Interface: An Overview", National Computer
15 Conference, 1982; Jeffrey J. Blatt, "A Primer on User Interface Patents," The Computer Lawyer, April 1992, Vol. 9 No. 4.

Research in interface design using metaphors in man-machine interfaces may be applied to computer controlled editing systems to incorporate and integrate
20 various production media resources, such as special effects, music, graphics and

the like. In the case of an editing system, a variety of resources must be integrated, scheduled and coordinated with one another to obtain a complete multi-media product. It is therefore desirable to allow a user to perform these functions, integration, scheduling and coordination, with
5 a minimum of operations. Further, it is desirable to allow the user to coordinate a variety of interrelated functions and to provide a display that allows the user to view the association between a function and the element the function is to affect.

These goals, minimizing operations, coordination and association,
10 are important for a wide variety of applications apart from the context of a video editor. Embodiments of the invention seek achievement of these goals for a video editor and a wide variety of other applications.

According to one aspect of the invention there is provided a method of providing a graphical user interface on a computer system,
15 said computer system including a CPU, a memory, and a graphics display screen, said method including the steps of:

- associating a function icon with a function;
- associating program data with a location on said graphics display screen;
- 20 displaying on said graphics display screen a representation of said function icon;

receiving a first data input from a user, said first data input selecting said function icon;

receiving a second data input from said user, said second data input causing said icon to be displayed on said graphics display screen at said
5 location; and

performing said function on said program data based upon the presence of said function icon at said location.

According to another aspect of the invention there is provided an apparatus for providing a graphical user interface on a computer system,
10 said computer system including a CPU, a memory, and a graphics display screen; said apparatus including:

first associating means for associating function icon with a function;

second associating means for associating program data with a
15 location on said graphics display screen;

display means for displaying on said graphics display screen a representation of said function icon;

first receiving means for receiving a first data input from a user, said first data input selecting said function icon;

20 second receiving means for receiving a second data input from said user, said second data input causing said icon to be displayed on said

graphics display screen at said location; and

function performing means for performing said function on said program data based upon the presence of said function icon at said location.

5 According to a further aspect thereof, the invention provides apparatus and methods for use in computer display systems, and in particular, a graphical user interface for use in computer systems. A display system including at least one central processing unit (CPU) is coupled through appropriate input/output (I/O) circuitry to input devices,
10 such as a cursor control device. The CPU is further coupled to a storage device for the storage of data and programs. The CPU is also coupled to a display device on which the user interface is displayed.

The user interface provided by a preferred form of implementation of the present invention includes function icons; for example a function
15 palette that displays a variety of function icons. Each function icon corresponds to a function that a user may wish to perform. The user selects a function icon from the palette corresponding to the desired function and the user places a copy of the icon at that point in the display where the function is to be performed. For example, in the context of
20 a multimedia video editor, the user may place an icon on a timeline that corresponds to source material. The desired function will then be

performed on the source material at that point where the user placed the icon, either immediately or upon subsequent command for the user.

The present invention has a very wide range of applications and may be used in conjunction with any type of computer program that
5 performs functions that may be represented by icons.

The invention will now be further described, by way of illustrative and non-limiting example, with reference to the accompanying drawings, in which:

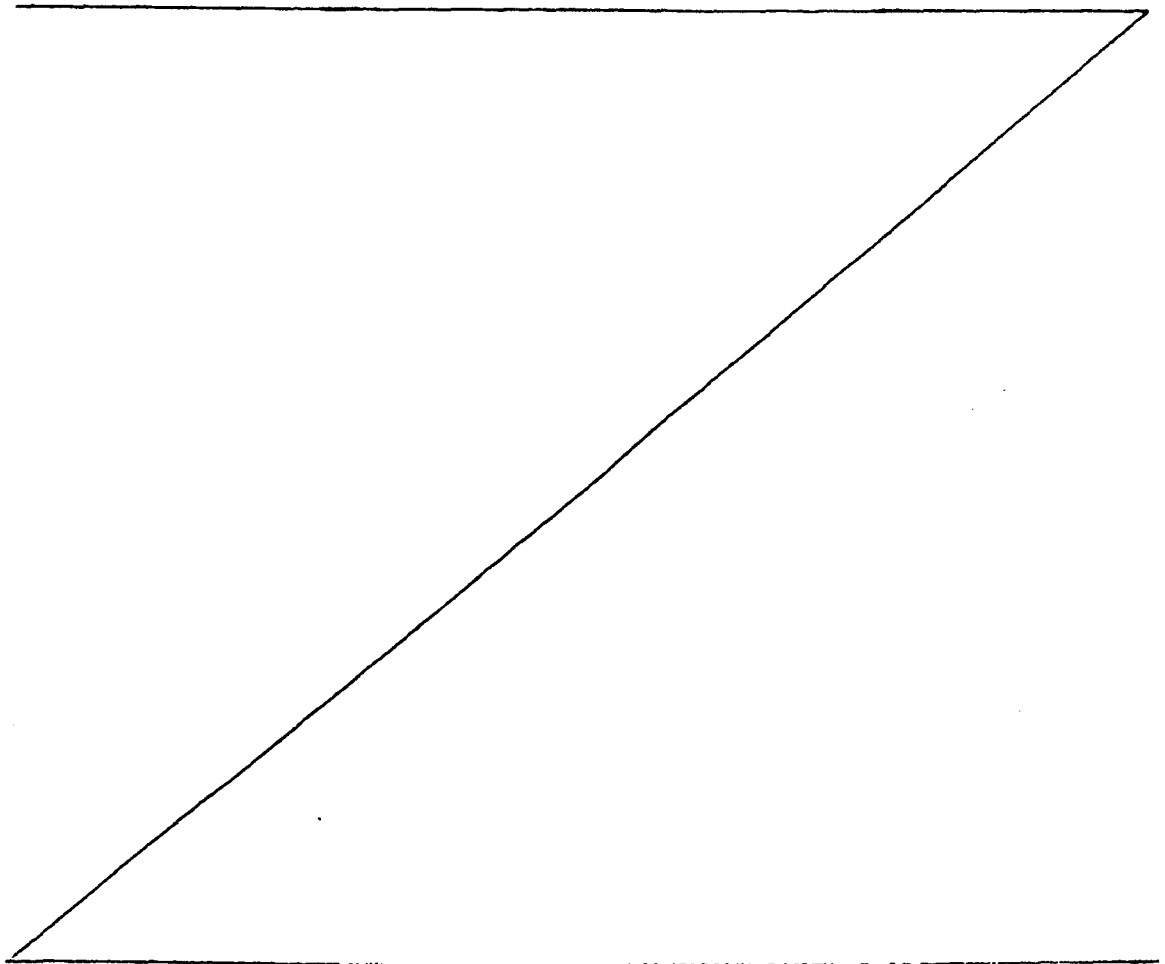


FIGURE 1 is a functional block diagram illustrating one possible computer display system incorporating the teachings of the present invention.

5 **FIGURE 2** illustrates one embodiment of the graphical user interface of the present invention in the context of a video editor.

FIGURE 3a illustrates a plurality of timelines that represent source material for a video editor.

10 **FIGURE 3b** illustrates the timelines of **Figure 3a** with a "synchronize" function icon placed at a particular location.

FIGURE 3c illustrates the result of the icons as displayed in **Figure 3b**.

FIGURE 4a illustrates the timelines of **Figure 3a** with "paste" and "split" icons placed at particular locations.

FIGURE 4b illustrates the result of the icons as displayed in **Figure 4a**.

FIGURE 5a illustrates the timelines of **Figure 3a** with a "split" icon placed within a single block.

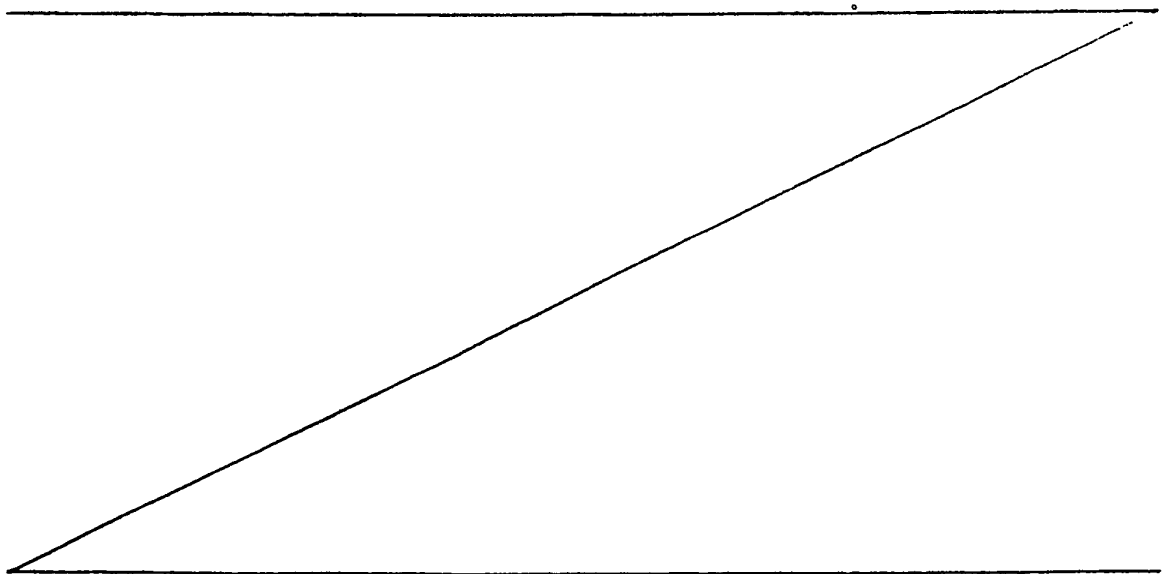
FIGURE 5b illustrates the result of the icons as displayed in **Figure 5a**.

5 **FIGURE 6a** illustrates the timelines of **Figure 3a** with "trim" icons placed at particular locations.

FIGURE 6b illustrates the result of the icons as displayed in **Figure 6a**.

10 **FIGURE 7a** illustrates a graphical user interface embodying the present invention for use in conjunction with an electronic spreadsheet.

FIGURE 7b illustrates the electronic spreadsheet of **Figure 7a** after a user has applied examples of the proxy icons of an embodiment of the present invention to the spreadsheet.



NOTATION AND NOMENCLATURE

The detailed descriptions which follow are presented largely in terms of algorithms and symbolic representations of operations on data bits within a computer memory. These descriptions and representations are the means used by those skilled in the data processing arts to most effectively convey the substance of their work to others skilled in the art.

An algorithm is here, and generally, conceived to be a self-consistent sequence of steps leading to a desired result. These steps are those requiring physical manipulations of physical quantities. Usually, though not necessarily, these quantities take the form of electrical or magnetic signals capable of being stored, transferred, combined, compared, and otherwise manipulated. It proves convenient at times, principally for reasons of common usage, to refer to these signals as bits, values, elements, symbols, characters, terms, numbers, or the like. It should be borne in mind, however, that all of these and similar terms are to be associated with the appropriate physical quantities and are merely convenient labels applied to these quantities.

Further, the manipulations performed are often referred to in terms, such as adding or comparing, which are commonly associated with mental operations

performed by a human operator. No such capability of a human operator is necessary, or desirable in most cases, in any of the operations described herein which form part of the present invention; the operations are machine operations. Useful machines for performing the operations of the present invention include
5 general purpose digital computers or other similar digital devices. In all cases there should be borne in mind the distinction between the method operations in operating a computer and the method of computation itself. The present invention relates to method steps for operating a computer in processing electrical or other physical signals to generate other desired physical signals.

10 The present invention also relates to apparatus for performing these operations. This apparatus may be specially constructed for the required purposes or it may comprise a general purpose computer as selectively activated or reconfigured by a computer program stored in the computer. The algorithms presented herein are not inherently related to a particular computer or other
15 apparatus. In particular, various general purpose machines may be used with programs written in accordance with the teachings herein, or it may prove more convenient to construct more specialized apparatus to perform the required method steps. The required structure for a variety of these machines will appear from the description given below.

The present invention discloses methods and apparatus for an improved graphical user interface. Although the present invention is described with reference to specific icons, windows and applications programs etc., it will be appreciated by one of ordinary skill in the art
5 that such details are disclosed simply to provide a more thorough understanding of the present invention. It will therefore be apparent to one skilled in the art that the present invention may be practiced without these specific details.

Referring to Figure 1, a hardware configuration used in
10 implementing an embodiment of the present invention is conceptually illustrated. Figure 1 illustrates a computer editing system for generating a window-based graphic user interface in accordance with the teachings of the present invention. As illustrated, the computer editing system includes a computer 20 which comprises four major components. The
15 first of these is an input/output (I/O) circuit 22, which is used to communicate information in appropriately structured form to and from other portions of the computer 20. In addition, computer 20 includes a central processing unit (CPU) 24 coupled to the I/O circuit 22 and to a memory 26. These elements are those typically found in most computers
20 and, in fact, computer 20 is intended to be representative of a broad category of data processing devices.

Also shown in **Figure 1** is a keyboard **30** for inputting data and commands into computer **20** through the I/O circuit **22**, as is well known. Similarly, a CD ROM **34** is coupled to the I/O circuit **22** for providing additional programming capacity to the system illustrated in **Figure 1**. It will be appreciated that
5 additional devices may be coupled to the computer **20** for storing data, such as magnetic tape drives, buffer memory devices, and the like. A device control **36** is coupled to both the memory **26** and the I/O circuit **22**, to permit the computer **20** to communicate with multi-media system resources. The device control **36** controls operation of the multi-media resources to interface the multi-media
10 resources to the computer **20**. For example, in a video editing environment, the computer **20** may be coupled through the device control **36** to a video tape recorder **40**, a music synthesizer **42**, an audio tape recorder **44**, a special effects resource **46**, and a disc player **41** as shown. The device control **36** may comprise an independent control and interface device or a software process for execution by
15 the CPU **24**.

A display monitor **50** is coupled to the computer **20** through the I/O circuit **22**. In the preferred embodiment, the display monitor **50** will comprise a high resolution color display with 16 or more bits of color per pixel but the display
20 monitor **50** may comprise an 8 bit color display. A cursor control device **52** includes switches **54** and **56** for signaling the CPU **24** in accordance with the

5 teachings of the present invention. Cursor control device 52 (commonly referred to a "mouse") permits a user to select various command modes, modify graphic data, and input other data utilizing switches 56 and 54. More particularly, the cursor control device 52 permits a user to selectively position a cursor 58 at any
10 desired location on a display screen 60 of the display 50. As illustrated in Figure 1, the cursor 58 is disposed with a window 65 in the present graphic user interface, to be described more fully below. Moreover, in the presently preferred embodiment, the present invention's window-based user interface is generated and displayed using software stored in either memories
15 26, 32 or CD ROM 34, and executed by the CPU 24.

In the presently preferred embodiment, the cursor control 52 utilizes well known apparatus and methods for signaling CPU 24 of position changes of cursor 58 by movement of the cursor control over a surface. However, it will be
20 appreciated by one skilled in the art that a variety of well known cursor control devices may be utilized by the present invention, including other control devices such as mechanical mice, trackballs, joy sticks, graphic tablets, other keyboard or so called virtual reality inputs and the like. The cursor control 52 in Figure 1 is intended to encompass all such equivalent devices.

Referring to **Figure 2**, a graphical user interface for a computer editing system configured in accordance with the present invention is illustrated. The computer **20** displays on the display screen **60** a graphical user interface that permits a user of the computer editing system to perform a number of re-editing functions. The graphical user interface displayed on the display screen **60** contains a plurality of tracks **76** and **78** that display timelines that include any number of source blocks **80**, **82**, **84**, **86**, **88**, **90**, **92**, **94**, **96** and **98** arranged in a sequence. The track **76** corresponds to video and the track **78** corresponds to audio. The source blocks **80**, **82**, **84**, **86**, **88**, **90**, **92**, **94**, **96** and **98** correspond to underlying source material that comprise the final, edited product. For example, the source block **80** may correspond to a video recording of a parachute jump from the ground while the block **84** may correspond to a video view from the perspective of the parachuter. An editor manipulates the source blocks to achieve a desired sequence.

The editor manipulates the source blocks by performing various functions such as cutting and splicing. The present disclosure describes an improved method and apparatus for performing these functions and almost any other type of functions. According to the present arrangement, a function palette **68** is displayed within the display screen **60**. The function palette **68** includes a plurality of icons

62, 64, 66, 70, 72 and 74, that represents functions that the user may wish to perform.

In one embodiment, the user implements the desired function by selecting an icon with the mouse 52. The user positions the cursor 58 on one of the icons 62, 64, 66, 70, 72 and 74 that represents a desired function and clicks a button 54 on the mouse 52 to select the item. With the button 54 depressed, the user drags a copy of the icon and places it on one of the timelines 76 or 78 where it is desired to perform the function. The copy of the icon is positioned on the display screen 60 where the user releases the button 54.

The icon function indicates that a corresponding function is to be performed at the point where the icon was placed. For example, an icon 82 may represent a "paste" function that indicates that the source blocks 80 and 82 are to remain attached. Thus, if the user moves the source block 82 to the right, the source block 80 will correspondingly move.

The memory 26 of computer 20 contains a program that causes the CPU 24 to detect a user's access to the palette 68. When one of the icons in the palette 68 is finally positioned on the display screen 60, the display address is provided to the computer 20 through an appropriate input/output routine. The address of the

icon is then compared with an address table stored in the memory 26 to determine what function the icon will perform. The function may be performed immediately or upon further user input.

5 If the function is to be performed upon further user input, a flag may be set corresponding to the type of user input that will trigger the function. When this type of input is received by the computer 20, the flag is checked and if it indicates that a function may correspond to that type of input, the computer then determines whether the input triggers the function based upon location of the icon, screen position of the movement and other factors that may be stored in a table. The
10 table may contain an address of the function to be performed that is accessed when the function is triggered. The function, in turn, may need to access the table to determine the results of the function. As will be appreciated by one of the skill in the art, there are other methods for performing functions according to the screen location of an icon.

15 The function to be performed by an icon may be modified by the positioning of the icon. For example, the function performed by an icon placed at the boundary of a block may be distinct from the function performed by an icon placed within a block. It will be appreciated that there are numerous adaptations of position dependent functionality for applications other than a multimedia editor.

Figures 3a-3c illustrate the operation of a synchronization function icon.

Figure 3a illustrates the tracks 76 and 78 comprised of a plurality of source blocks 100, 102, 104, 106, 108 and 110. The tracks 76 and 78 are displayed on the display screen 60. The source block 102 may correspond to video footage of two chimpanzees fighting over a banana while the source block 108 may correspond to the audio of the fighting chimpanzees. The editor may wish to synchronize the video source block 102 with the audio source block 108.

Figure 3b illustrates the use of a function icon to synchronize the source blocks 102 and 108. The user positions an icon 112 on points within both the blocks 102 and 108 through the use of the mouse 52, as previously described. As will readily be appreciated, the icon 112 may comprise separate icons and also may comprise any shape and be positioned in a manner other than is shown in Figure 3b.

Figure 3c illustrates the result of the function icon 112 after the user has moved one of the blocks 102 or 108. If the user has moved the block 102 to an earlier point on the track 76, corresponding to a leftward movement, the icon 112 ensures that the block 108 moves along with the source block 102. Similarly, if the user moves the source block 108, the icon 112 ensures that the block 102

moves correspondingly. Thus, the synchronization icon ensures that the video block 102 and the audio block 108 remain synchronized.

Figures 4a-4b illustrate "paste" and "split" icons applied to the tracks 76 and 78 as illustrated in Figure 3a. In one embodiment, to ensure that the block 102 and the block 104 move together, the user places a paste "icon" on both the block 102 and the block 104. In alternate embodiments, the user may place a "paste" icon 120 on the boundary of either of the blocks 102 or 104 and an arrow may indicate the direction of the "paste." Many other alternatives are possible. Similarly, to ensure that the block 110 does not move with the block 108, an icon 122 is placed on the boundaries of the blocks 108 and 110 as illustrated. In practice, a default will dictate whether a block moves with an adjoining block and only the "paste" icon 120 or the "split" icon 122 may be necessary. Figure 4a also illustrates a synchronization icon 112 that has been placed on blocks 102 and 108 to ensure that these blocks move together, as previously described.

Figure 4b illustrates the results of the "paste" icon 120 and the "split" icon 122 in conjunction with the synchronization icon 112, after the source blocks 102 and 108 have been moved. The user has provided inputs that move the source blocks 102 to the left and icon 112 causes block 108 to move synchronously with the block 102. It will be appreciated that the source blocks 102 and 108 may also

move independently. As shown, the source block 104 has moved along with the source block 102 while the source block 110 has remained in place. Another split icon 124 may be displayed on the block 110 after the movement of the source block 108.

5 **Figures 5a-5b** illustrate similar operations similar to those shown in **Figures 4a-4b** except with the "split" icon 122 applied to a single block 121. A split icon within a single block splits the block when a part of the block is moved. The interface may allow the user to move only part of a block or an entire block and if the user moves that part of the block to the left of icon 122, the block 121
10 will split into pieces according to the placement of icon 122. **Figure 4b** illustrates the results of the "split" icon 122 after the left of the source block 121 has been moved. As shown, the block 121 has been split into blocks 108 and 110. Had the user moved the entire block 121, as opposed to part of the block 121, the block 121 would not have split. Another split icon 124 may be displayed on the
15 block 110 after the movement of the source block 108.

Figures 6a and 6b illustrate the use of "trim" icons applied to the tracks 76 and 78 as illustrated in **Figure 3a**. As illustrated in **Figure 6a**, a low priority trim icon 130 is placed on the source block 102 and a high priority trim icon 132 is placed on the source block 108. When the user moves a source block over a

different source block, the priority of the trim icon indicates which source block will be trimmed. In practice, a default will dictate whether a moving block or a stationary block will be trimmed and thus only the low priority icon 130 or the high priority icon 132 may be necessary. The user may adjust the default depending upon which setting is more efficient.

Figure 6b illustrates the results of the low priority trim icon 130 and the high priority trim icon 132. When the block 102 was moved to the right, its source material was trimmed while the source material of the block 110 was trimmed when the block 108 was moved to the right.

The present invention has a variety of applications in contexts other than video editors. **Figure 7a** illustrates a spreadsheet 150 configured according to the graphic user interface of the present invention and displayed on the display screen 60. A function palette 152 includes a plurality of function icons, including "increment right" icon 154, "increment down" icon 156, "substitution icon" 158, "increment both" icon 160. The icons may be placed next to cell indicators that occur in formulae within a cell. For example, cell A6 is set equal to "A1 * A2" and the increment right icon 154 is placed next to "A1" as shown while the increment both icon 160 is placed next to "A2" as shown.

In one embodiment, for the "increment right" icon 154, the spreadsheet automatically fills in any blank cells to the right of cell A6 with the value of A1 incremented by the number of columns to the right. Thus, for example, cell B6 would contain A1 incremented by one column for a value of B1 while cell C6 would be incremented by 2 columns for the value of C1. The "increment down" icon 156 has similar operation except that the row numbers are incremented instead of the columns. The increment both icon increments both the row and the column. In an alternate embodiment, the values are incremented after the user clicks on the cell A6 and drags the cursor to a different cell.

Figure 7b illustrates the value in the cells after the user has dragged the cursor to cell C8. As shown in the figure, the value A2 in cell A6 has been incremented both right and down while the value A1 in cell A6 has been incremented to the right. Thus, cell B6 contains the formula "B1 * B2", cell C6 contains the formula "C1 * C2", cell B7 contains the formula "B1 * B3" and the remaining cells contain the appropriate formulae as shown.

The "substitution icon" 158 substitutes the current cell value where the default is to transfer the same formula terms. For example, as illustrated in Figure 7a, cell AN contains the value "A1 + A2" and the "substitution icon" 158 placed next to "A1" ensures that the value in cells to the right of AN are

substituted for "A1." As shown in Figure 7b, cell BN contains the formula "BN + A2" where BN has been substituted, according to the "substitution icon" 158, for "A1" and "A2" has been copied from cell AN, which is the default procedure.

5 The "substitution icon" 158 is not required if substitution is the default when cells are copied. However, if substitution is the default, then a "hold constant" icon may be used to alter the default of substitution. Similarly, "hold constant right" and "hold constant down" icons may be used as the complementary icons for the "increment right" icon 154 and the "increment down" icon 156.

10 The present invention may also be used in conjunction within a data processing system such as stock control records in a warehouse management system. In this case, the proxy icons may represent processing functions on datasets within the data processing system. For example, count, merge, replace, sort, move, sum and select icons may be employed and the functions might display a result when they are executed. For example, the sum of
15 the values in the datasets may be displayed after a sum icon is selected by the user. The user may add icons to the model to show the processes that occur on each dataset in response to the triggering function.

As yet another example of the application of the present invention, proxy icons may be used in conjunction with a computer drawing package. Proxy icons may be attached to faces or vertices of graphical objects to indicate how the object should transform in various circumstances. Exemplary icons may include

5 "vertical constrain," "horizontal constrain," "stretch" and "stay attached to adjacent" icons. Many other icons are possible.

Although the present invention has been described in terms of a preferred embodiment, it will be appreciated that various modifications and alterations might be made by those skilled in the art without departing from the scope of

10 the invention. The present invention is applicable to a wide variety of applications and should therefore be measured in terms of the claims which follow.

CLAIMS

1 1. A method of providing a graphical user interface on a computer system,
2 said computer system including a CPU, a memory, and a graphics display screen,
3 said method including the steps of:

4 associating a function icon with a function;
5 associating program data with a location on said graphics display screen;
6 displaying on said graphics display screen a representation of said function
7 icon;
8 receiving a first data input from a user, said first data input selecting said
9 function icon;
10 receiving a second data input from said user, said second data input causing
11 said icon to be displayed on said graphics display screen at said location; and
12 performing said function on said program data based upon the presence of
13 said function icon at said location.

1 2. The method of claim 1 wherein said program data comprises source data to
2 be edited in a computer editing system.

1 3. The method of claim 2 wherein said source data comprises a block of data
2 that is to be edited with other blocks of data.

1 4. The method of claim 3 wherein at least a part of said block of data is
2 displayed at said location such that said icon is displayed on said block of data.

1 5. The method of claim 4 wherein said function comprises a paste function
2 such that when said block is moved, an adjacent block moves along with said
3 block.

1 6. The method of claim 4 wherein said block corresponds to a first track and
2 said function comprises a synchronization function such that when said block is
3 moved, a different block from an adjoining track moves along with said block.

1 7. The method of claim 4 wherein said function comprises a trim function
2 such that when said block is moved, an adjacent block is overwritten by said
3 block.

1 8. The method of claim 1 wherein said program data comprises a formula in a
2 spreadsheet cell.

1 9. The method of claim 1 further comprising of displaying a palette including
2 said representation of said function icon.

1 10. The method of claim 9 wherein said representation of said function icon
2 comprises a copy of said function icon.

1 11. The method of claim 10 wherein said first data input comprises input from
2 a mouse.

1 12. The method of claim 11 wherein said second data input comprises input
2 from a mouse, said data input causing a copy of said function icon to be dragged
3 to said location.

1 13. The method of claim 1 wherein at least of part of said program data is
2 displayed on said graphics display screen at said location.

1 14. The method of claim 1 wherein said step of performing said function
2 further comprises the step of automatically performing said function.

1 15. The method of claim 1 wherein said step of performing said function
2 further comprises the step of performing said function after receiving further input
3 from said user.

1 16. An apparatus for providing a graphical user interface on a computer
2 system, said computer system including a CPU, a memory, and a graphics display
3 screen, said apparatus including:

4 first associating means for associating a function icon with a function;

5 second associating means for associating program data with a location on
6 said graphics display screen;

7 display means for displaying on said graphics display screen a
8 representation of said function icon;

9 first receiving means for receiving a first data input from a user, said first
10 data input selecting said function icon;

11 second receiving means for receiving a second data input from said user,
12 said second data input causing said icon to be displayed on said graphics display
13 screen at said location; and

14 function performing means for performing said function on said program
15 data based upon the presence of said function icon at said location.

1 17. The apparatus of claim 16 wherein said program data comprises source data
2 to be edited in a computer editing system.

1 18. The apparatus of claim 17 wherein said source data comprises a block of
2 data that is to be edited with other blocks of data.

1 19. The apparatus of claim 18 wherein at least a part of said block of data is
2 displayed at said location such that said icon is displayed on said block of data.

1 20. The apparatus of claim 19 wherein said function comprises a paste function
2 such that when said block is moved, an adjacent block moves along with said
3 block.

1 21. The apparatus of claim 19 wherein said block corresponds to a first track
2 and said function comprises a synchronization function such that when said block
3 is moved, a different block from an adjoining track moves along with said block.

1 22. The apparatus of claim 19 wherein said function comprises a trim function
2 such that when said block is moved, an adjacent block is overwritten by said
3 block.

1 23. The apparatus of claim 16 wherein said program data comprises a formula
2 in a spreadsheet cell.

1 24. The apparatus of claim 16 further comprising of displaying a palette
2 including said representation of said function icon.

1 25. The apparatus of claim 24 wherein said representation of said function icon
2 comprises a copy of said function icon.

1 26. The apparatus of claim 25 wherein said first data input comprises input
2 from a mouse.

1 27. The apparatus of claim 26 wherein said second data input comprises input
2 from a mouse, said data input causing a copy of said function icon to be dragged
3 to said location.

1 28. The apparatus of claim 16 wherein at least of part of said program data is
2 displayed on said graphics display screen at said location.

1 29. The apparatus of claim 16 wherein said function performing means further
2 includes means for automatically performing said function.

30. The apparatus of claim 16 wherein said function performing means further includes means for performing said function after receiving further input from said user.

5 31. A method of providing a graphical user interface on a computer system, the method being substantially as herein described with reference to the accompanying drawings.

32. An apparatus for providing a graphical user interface on a
10 computer system, the apparatus being substantially as herein described with reference to the accompanying drawings.



Application No: GB 9606467.0
Claims searched: ALL

Examiner: R. F. King
Date of search: 25 June 1996

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.O): H4T[TBLA, TBLC, TBLM, TBLX]

Int Cl (Ed.6): G06F 3/023, 3/033, 3/037

Other: ONLINE: WPI; COMPUTER

Documents considered to be relevant:

Category	Identity of document and relevant passage	Relevant to claims
X	GB2 254 989 A (TOSHIBA KK) See whole doc.	1, 13, 14
A	EP 0 570 137 A1 (IBM) See abstract	1
A	EP 0 468 911 A2 (IBM) See abstract	1
A	US 5,422,993 (IBM) See abstract	1

X Document indicating lack of novelty or inventive step
Y Document indicating lack of inventive step if combined with one or more other documents of same category.
& Member of the same patent family

A Document indicating technological background and/or state of the art.
P Document published on or after the declared priority date but before the filing date of this invention.
E Patent document published on or after, but with priority date earlier than, the filing date of this application.

